



# Integração de componentes em páginas Web

## Objetivos:

- Componentes de *Twitter Bootstrap*.
- Gráficos usando *Highcharts JS*.
- Mapas e elementos georeferenciados.
- Sistemas de gestão de comentários.

## 8.1 Páginas Web

As páginas Web atuais necessitam de possuir uma estrutura e estilos muito sofisticados. Cada vez mais a interação faz-se através de interfaces Web e menos através de aplicações que os utilizadores necessitem de instalar no seu computador.

Bons exemplos desta tendência são aplicações como o Google Mail [1], o Google Docs [2] ou o Microsoft Office 365 [3].

Como vimos na aula dedicada ao tema de páginas Web, existem componentes que permitem acelerar o desenvolvimento, fornecendo, de uma maneira simplificada, estrutura e estilo agradável. Estes sistemas permitem ainda melhorar a interação através da inclusão de componentes "pré-feitos" que aumentam a atratividade e funcionalidade de uma página.

Isto reforça a ideia de que o desenvolvimento de um sistema deve favorecer componentes já existentes, disponibilizados por outras entidades, não devendo o programador pretender desenvolver toda a programação e estilo. Só em casos muito específicos se pode tomar esta atitude, à custa de um maior esforço de desenvolvimento e de manutenção do código desenvolvido.

As páginas Web são um dos domínios onde mais se utilizam recursos externos, sendo o resultado visível na quantidade e complexidade das páginas disponíveis na *Internet*. Este guião irá abordar alguns componentes que podem ser utilizados para enriquecer uma página Web, com foco em componentes dinâmicos para a representação de gráficos, mapas e interação com visitantes.

Para tal, irá fazer-se uso de uma página construída usando o *Twitter Bootstrap*, sendo que serão adicionados componentes de forma a enriquecer as suas funcionalidades.

Relembra-se que uma página mínima, fazendo uso de *Twitter Bootstrap* pode ser criada fazendo inclusão da página de estilos correta. De forma a suportar um modelo de interação mais rico, poderá também ser adicionado um conjunto de *scripts*.

O exemplo que se segue resulta numa página muito minimalista com *Twitter Bootstrap* mas que poderá servir de base para o restante guião:

---

```
<!DOCTYPE html>
<html lang="pt">
  <head>
    <meta charset="utf-8">
    <title>Integração de Componentes em Páginas WEB</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- Bootstrap minified CSS -->
    <link rel="stylesheet"
href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">
    <!-- Optional theme -->
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap-theme.min.css">
    <!-- jQuery library -->
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
    <!-- Bootstrap JavaScript -->
    <script
src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>
  </head>

  <body>
    <div class="container">
      (Conteúdo da página.)
    </div>
  </body>
</html>
```

---

De realçar que ao contrário do exemplo fornecido numa aula anterior, neste caso a página inclui componentes de Cascading Style Sheets (CSS)[4] e de JavaScript (JS)[5]. Existe igualmente um elemento `<div>` com a classe `container`, no qual se deverá incluir o conteúdo da página.

### Exercício 8.1

No diretório da disciplina crie um subdiretório chamado **web** onde deve executar os exercícios deste guião.

Entre no diretório recém criado e crie uma página com o conteúdo do exemplo anterior. Verifique o funcionamento correto desta página.

## 8.2 Componentes

Uma página *Web* é composta por várias secções, definindo a sua estrutura, sendo que em cada secção existirão componentes variados tais como menus, sub-menus, painéis, imagens, entre outros. A linguagem HyperText Markup Language (HTML)[6] fornece já um grande conjunto de ferramentas que possibilitam a criação de páginas ricas, no entanto por vezes é insuficiente aplicar as marcas de forma isolada, sendo necessário conjugar a marca (HTML), um estilo (CSS) e ações (JS).

Para uma lista completa dos componentes disponíveis para *Twitter Bootstrap*, consultar <http://getbootstrap.com/components>. Neste guião será abordado um número reduzido de componentes.

### 8.2.1 Menus e Sub-Menus

Os menus e sub-menus são vitais para a navegação numa página. Tipicamente estes são implementados através do recurso a marcas `<ul>` e `li`. Ou seja, um menu é tratado como uma lista de items. O estilo aplicado a essa lista é que lhe atribui o aspeto típico de um menu. Usando *Twitter Bootstrap*, o menu principal de uma página chama-se barra de navegação e é incluída dentro de elementos `<nav>` e `<div>`.

```
<nav class="navbar navbar-default">
  <div class="navbar-header">
    <a class="navbar-brand" href="#">Engenharia Informática</a>
  </div>

  <div class="navbar-collapse collapse">
    <ul class="nav navbar-nav">
      <li class="active"><a href="#">Tópico A</a></li>
      <li><a href="#">Tópico B</a></li>
    </ul>

    <ul class="nav navbar-nav navbar-right">
      <li class="active"><a href="#">Default</a></li>
    </ul>
  </div>
</nav>
```

O elemento `<nav>` inicial possui classes (`navbar` e `navbar-default`) que servem para definir o posicionamento e o aspeto (borda, sombra).<sup>1</sup> Este elemento inclui marcas que indicam o conteúdo, respetivamente: uma `<div>` com classe `navbar-header` indicando o nome da página; outra `<div>` com classes para definirem estilo; duas marcas `<ul>` para a construção do menu em si. Cada elemento (`<li>`) da lista será uma opção do menu.

### Exercício 8.2

Adicione uma barra de navegação à sua página. A barra deve ser incluída no início do elemento com classe `container`.

Para obter sub-menus é necessário utilizar a classe `dropdown` num item do menu. Os elementos do sub-menu são implementados através de uma nova lista.

```
<li class="dropdown">
  <a href="#" class="dropdown-toggle" data-toggle="dropdown">Tópico C<b class="caret"></b></a>
  <ul class="dropdown-menu">
    <li><a href="#">Opção1</a></li>
    <li class="divider"></li>
    <li><a href="#">Opção2</a></li>
  </ul>
</li>
```

### Exercício 8.3

Adicione o exemplo acima à sua página.

O resultado deverá ser o apresentado na Figura 8.1.

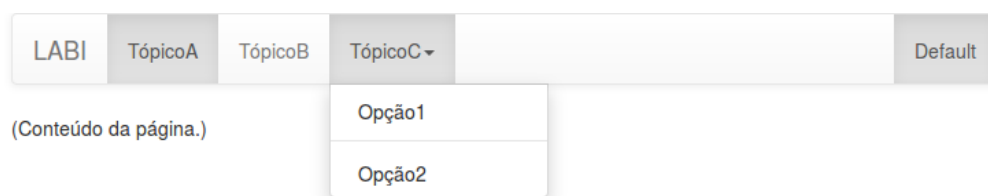


Figura 8.1: Barra de navegação

De forma a navegar na página, é possível criar ligações internas. Estas ligações funcionam através do atributo `id` de cada elemento. Por exemplo, se existir um elemento com `id=tópicoA`, uma marca `<a>` pode enviar o utilizador para esse elemento tendo `#tópicoA` no seu atributo `href`.

<sup>1</sup>A marca `<nav>` foi introduzida no HTML5. Em versões anteriores poderá usar-se uma marca `<div class="..." role="navigation">`.

O exemplo seguinte demonstra este caso.

```
<a href="#topicoA">Topico A</a>
...
<h2 id="topicoA">Este é o Tópico A</h2>
```

### Exercício 8.4

Na sua página, adicione marcas `<h2>` para cada um dos tópicos deste guião. Crie um número igual de ligações na barra de navegação, possibilitando ir rapidamente para cada tópico a partir desta barra. **Execute os restantes exercícios por baixo do tópico correspondente.**

## 8.2.2 Popups

Os *Popups* são bastante úteis para apresentar mensagens importantes aos utilizadores. Tipicamente sobrepõem-se à página e necessitam de uma ação explícita para desaparecerem, tal como clicar num botão. Usando *Twitter Bootstrap*, os *Popups* são implementados com recurso à classe `modal` aplicado à marca `<div>`. Tal como demonstrado no exemplo que se segue, existe uma marca `<div>` inicial da classe `modal` que depois possui diversas outras marcas `<div>` com cada uma das áreas do elemento. Pode identificar um cabeçalho (`modal-header`), um corpo (`modal-body`) e um rodapé (`modal-footer`). Note também a existência de duas marcas `button` correspondendo ao botão com o símbolo **X**, que tipicamente existe no topo de uma janela, e ao botão com o texto *Fechar*.

```
<h2 id="topicoA">Este é o Tópico A - PopUp</h2>
<div id="oMeuPopup" class="modal fade">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button
          type="button" class="close" data-dismiss="modal" aria-hidden="true">&times;
        </button>
        <h4 class="modal-title">Título</h4>
      </div>
      <div class="modal-body">
        <p>Conteúdo do PopUp</p>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-default" data-dismiss="modal">Fechar</button>
      </div>
    </div>
  </div>
</div>
```

Existem vários outros atributos, tais como **data-dismiss**, ou **aria-hidden** que servem para adicionar funcionalidades. O resultado poderá ser o apresentado na Figura 8.2.

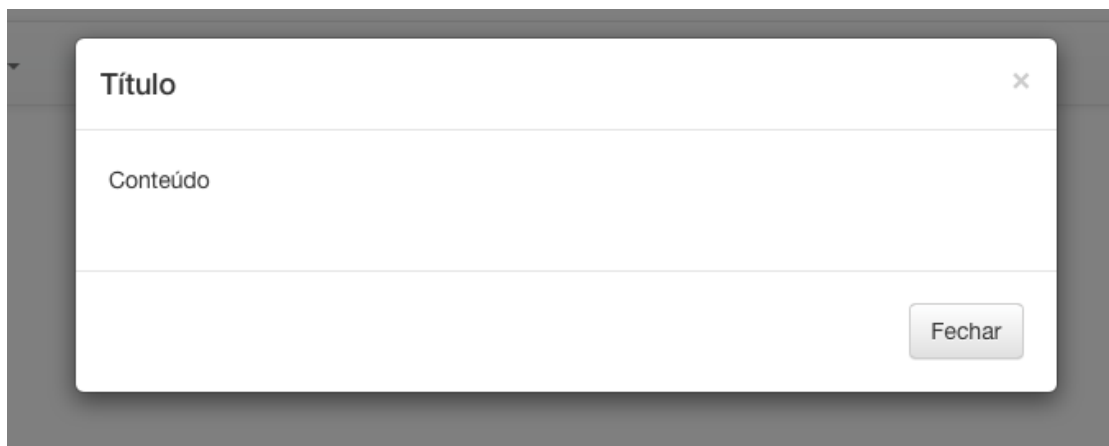


Figura 8.2: *Popup*

### Exercício 8.5

Adicione o exemplo acima. Repare que nada é apresentado. No entanto pode ver no código fonte que isto se deve apenas a um atributo do estilo.

Os *Popups* não são sempre apresentados apesar do seu conteúdo permanecer na página. Uma maneira simples de ativar o *Popup* será através de um botão, usando o atributo **data-target**. O valor deste atributo terá de corresponder ao valor do atributo **id** do *Popup* a apresentar.

```
<button class="btn btn-primary" data-toggle="modal" data-target="#oMeuPopup">  
  Lançar popup  
</button>
```

### Exercício 8.6

Implemente um exemplo com um botão que ativa um *Popup* e adicione-o à sua página. Verifique que o *Popup* já é apresentado.

## 8.3 Gráficos

O HTML5 suporta gráficos através dos elementos `<canvas>` e `<svg>`. No entanto, há recursos externos de JS que facilitam muito a criação de gráficos complexos. É o caso da biblioteca *Highcharts JS* que iremos introduzir nesta secção.

Para adicionar um gráfico com esta biblioteca é preciso:

1. Incluir, na marca **head**, os recursos de JS que permitem desenhar gráficos. No caso do *Highcharts JS* é necessária uma linha:

---

```
<!-- Highcharts JavaScript -->
<script type="text/javascript" src="http://code.highcharts.com/highcharts.js"></script>
```

---

2. Incluir na marca **body** da página (por exemplo no tópico B) um elemento que irá conter o gráfico.

---

```
<h2 id="topicoB">Este é o Tópico B - Gráficos</h2>
<div id="grafico-linhas" style="width: 400px; height: 300px;"></div>
```

---

Neste caso, o gráfico terá 400px por 300px de dimensão. O atributo `id` permitirá identificar o elemento nos scripts.

3. Incluir na marca **head** da página o script:

---

```
<script type="text/javascript" src="grafico-linhas.js"></script>
```

---

cujo conteúdo (ficheiro JS `grafico-linhas.js`) deve configurar o gráfico e os dados:

---

```
function desenhaGrafico() {
    $("#grafico-linhas").highcharts({
        chart: { type: "line" },
        title: { text: "Média de temperaturas" },
        xAxis: { categories: ["Jan", "Fev", "Mar", "Abr", "Mai", "Jun",
                            "Jul", "Ago", "Set", "Out", "Nov", "Dez"] },
        series: [
            { name: "Lisboa",
              data: [7.0, 6.9, 9.5, 14.5, 18.2, 21.5, 25.2, 26.5, 23.3, 18.3, 13.9, 9.6] },
        ]
    });
};
```

---

4. Invocar a função de JS que desenha o gráfico (a seguir ao elemento do ponto 2).

---

```
<button class="btn btn-primary" onclick="desenhaGrafico()">Grafico de Linhas</button>
```

---

Ao compor este exemplo, o resultado deverá ser semelhante ao apresentado na Figura 8.3.

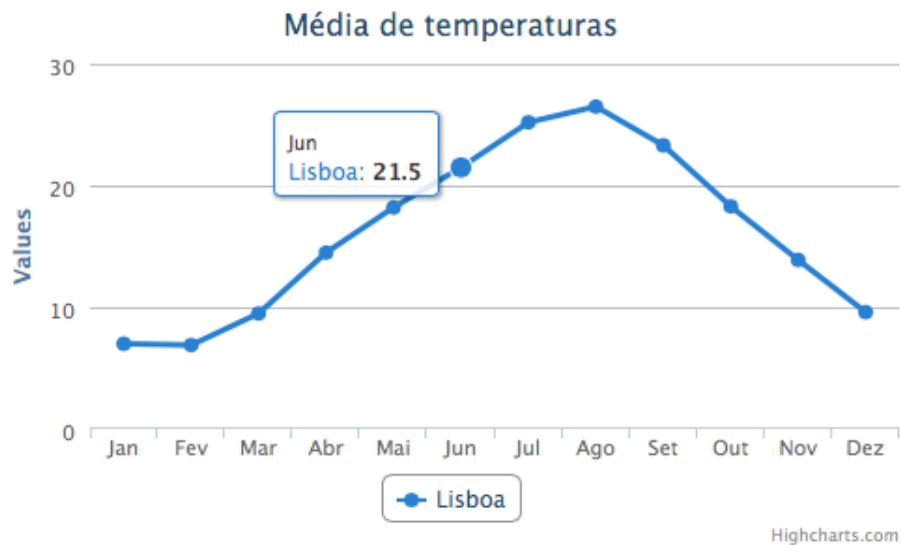


Figura 8.3: Gráfico de linhas

### Exercício 8.7

Adicione um gráfico de linhas com as médias de temperatura do sítio onde cresceu. Pode consultar esta informação no endereço <http://www.weatherbase.com>.

Repare que no exemplo anterior apenas é mostrada a série de temperaturas relativas a Lisboa. No entanto é possível ter várias séries simultaneamente. Para isso basta que o campo **series** possua o seguinte formato:

```
series: [  
  {  
    name: "nome-da-serie-1",  
    data: [... valores ...]  
  },  
  {  
    name: "nome-da-serie-2",  
    data: [... valores ...]  
  }  
]
```

### Exercício 8.8

Adicione uma segunda linha com as temperaturas de Aveiro de forma a comparar os dois locais.



A biblioteca *Highcharts JS* possibilita muitos outros tipos de gráficos, tais como: **pie**, **column**, **scatter**, **bar**, **area** [7]. Frequentemente, modificar a aparência do gráfico apenas requer alterar o seu tipo, definido da seguinte forma:

```
$("#grafico-colunas").highcharts({  
  chart: { type: "column" },  
  title: { text: "Média de temperaturas" },  
  ...
```

### Exercício 8.9

Faça cópias do ficheiro com a definição do gráfico e altere-as para testar os vários tipos de gráficos possíveis. Inclua todos os ficheiros na sua página. Adicione um botão e um elemento para desenhar cada gráfico.

Em alternativa pode criar um único ficheiro **graficos.js**, desenvolver várias funções com nomes diferentes (**desenhaGráficoLinhas()**, **desenhaGráficoColunas()**, ...) consoante o tipo de gráfico pretendido, não se esquecendo de alterar o identificador ("**#grafico-tipo**") de cada um deles na respetiva função e no respetivo botão.

## 8.4 Mapas

Outro componente muito comum em páginas *Web* são os mapas ou outros elementos que apresentam informação geo-referenciada. Os mais populares são provavelmente o *Google Maps* [8] e o *OpenStreetMap* [9].

A utilização destes elementos é análoga à dos gráficos, visto que é necessária a inclusão de um recurso externo, a existência de um elemento onde apresentar o mapa e algum código JS para personalizar o conteúdo.

Apesar de ser possível utilizar serviços como o *Google Maps* diretamente, por vezes é mais vantajoso utilizar bibliotecas que facilitem a integração e utilização. Uma das vantagens é a possibilidade de utilizar mapas de diversos fornecedores sem modificações relevantes na programação. Neste guião recomendamos a utilização da biblioteca *Leaflet JS* [10].

Para utilizar *Leaflet JS* é necessário efetuar os seguintes passos:

1. Incluir a definição de estilos e de código do *Leaflet JS*.

```
<!-- Leaflet CSS e JavaScript -->  
<link rel="stylesheet" href="https://unpkg.com/leaflet@1.0.2/dist/leaflet.css">  
<script src="https://unpkg.com/leaflet@1.0.2/dist/leaflet.js"></script>
```

2. Criar um elemento na marca **body** da página (por exemplo no tópico C - Opção1) para o mapa e incluir um atributo **id** para o identificar no código JS.

```
<h2 id="topicoC1">Este é o Tópico C - Mapas</h2>
<div id="oMeuMapa" style="width: 500px; height: 400px"></div>
```

3. Finalmente é necessário incluir logo a seguir o código JS que cria o mapa.

```
<script type="text/javascript" src="map.js"></script>
```

Sendo que o conteúdo do ficheiro **map.js** poderá ser o seguinte:

```
var map = new L.Map("oMeuMapa", {center: [40.633258,-8.659097],zoom: 15});

var osmUrl="http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png";
var osmAttrib="Map data OpenStreetMap contributors";
var osm = new L.TileLayer(osmUrl, {attribution: osmAttrib});

map.addLayer(osm);
```

Neste caso o valor 40.633258 refere-se à latitude, o valor -8.659097 refere-se à longitude e o valor 15 é o nível de *zoom*.

O resultado deverá ser semelhante ao apresentado na Figura 8.4.



Figura 8.4: Mapa utilizando *Leaflet JS*

### Exercício 8.10

Adicione um mapa à sua página. Este mapa deverá mostrar a cidade de Aveiro e a cidade onde cresceu.

É possível associar funções a eventos que ocorram no mapa criado. O evento `click` é muito útil porque fornece as coordenadas de um ponto selecionado no mapa.<sup>2</sup>

Por exemplo, para mostrar as coordenadas atuais poderíamos incluir um elemento `<span id="coordenadas"></span>` na página e acrescentar a instrução

```
map.on("click", mostraCoordenadas);
```

bem como a definição da respetiva função ao código JavaScript (ficheiro `map.js`):

```
function mostraCoordenadas(e){  
    var s = document.getElementById("coordenadas");  
    s.innerHTML = "Latitude, Longitude = "+e.latlng.lat+", "+e.latlng.lng;  
}
```

Uma das utilidades dos mapas é a apresentação de pontos (ou marcadores), indicando a localização de um ponto de interesse. Esta funcionalidade é muito útil para localizar eventos, locais de interesse ou a morada de empresas e serviços.

Adicionar um ponto a um mapa requer que se defina as coordenadas do ponto e de seguida que se adicione este ponto ao mapa. No exemplo seguinte é criado um array com vários pontos, que são depois adicionados ao mapa.

```
var pontos = [  
    L.marker([40.633258, -8.659097]),  
    L.marker([40.642729, -8.747899])  
];  
  
for(i in pontos) {  
    pontos[i].addTo(map);  
}
```

Também pode ser adicionado um pequeno texto que será apresentado quando o apontador clicar em cima do ponto. Para isto, é utilizado o método `bindPopup(msg)` em que o argumento `msg` pode ser qualquer texto ou mesmo código HTML.

<sup>2</sup>Para a lista completa de eventos, consultar <http://leafletjs.com/reference-1.0.0.html#map-event>.

```
...  
L.marker([40.633258, -8.659097]).bindPopup("LABI@DETI")  
...
```

Por vezes é útil ajustar a vista de forma a contemplar todos os pontos adicionados. Aqui é necessário criar um grupo com todos os pontos e depois invocar um método que ajusta a vista de forma automática.

```
...  
var grupo = new L.featureGroup(pontos);  
map.fitBounds(grupo.getBounds());
```

### Exercício 8.11

Adicione marcadores que identifiquem vários locais de interesse para si (morada em Aveiro, morada permanente, onde estuda, etc). Não se esqueça de adicionar textos que descrevam os locais e de ajustar o mapa aos pontos.

Frequentemente é desejável apresentar imagens diferentes dependendo da natureza do local assinalado. Isto é simples pois os elementos **marker** podem ser criados com uma lista de opções, onde se inclui o ícone.<sup>3</sup>

```
var iconeUA = L.icon({ iconUrl: "http://xcoa.av.it.pt/ua.png" });  
...  
L.marker([40.633258, -8.659097], {icon: iconeUA}).bindPopup("LABI@DETI")  
...
```

### Exercício 8.12

Adicione imagens personalizadas aos seus ícones. Na página <http://mapicons.nicolasmollet.com> encontra muitos e variados ícones. Terá de escolher os ícones e colocá-los junto com a sua página.

Além de marcadores também é possível adicionar polígonos que sinalizam uma área alargada e não um ponto individual. A metodologia é semelhante à criação de marcadores, mas os polígonos são criados através de um array de pontos.

<sup>3</sup>Para uma lista completa de opções, consultar <http://leafletjs.com/reference-1.0.0.html#marker>.

A Reitoria de Universidade de Aveiro pode ser indicada através de um polígono, da seguinte forma:

```
var reitoria = L.polygon(  
  [ [40.63102, -8.65793], [40.63149, -8.65731],  
    [40.63126, -8.65699], [40.63078, -8.65759] ],  
  { color: "red" } );  
reitoria.addTo(map);
```

### Exercício 8.13

Adicione um polígono que delimite o DETI.

Ao polígono que criou, use o método **bindPopup(msg)** para mostrar informação do departamento quando se clicar no polígono.

### Exercício 8.14

Em vez de utilizar o tipo **polygon**, se utilizar o **polyline** o resultado será uma linha e não um polígono. Utilize estes dois tipos de objetos para indicar onde mora e qual o trajeto para a Universidade de Aveiro.

## 8.5 Comentários

Um aspeto interessante de se adicionar a um blog ou outra página com visitantes é a capacidade de se incluir comentários. Existem várias maneiras de obter esta funcionalidade, sendo que a mais simples é a adição de um componente externo em JS.

Um exemplo é o serviço Disqus [11], que apenas necessita da inclusão de código HTML e JS na página em questão.

### Exercício 8.15

De forma a adicionar comentários à sua página, aceda a <http://disqus.com> e identifique-se com o utilizador **alunolabi** e a palavra passe **alunolabi**.

Seguidamente, aceda a <http://disqus.com/admin/universalcode/> de forma a obter o código a incluir na página. O valor da variável **disqus\_shortcode** deverá ser **labi**.

**Nota:** Por favor não altere os dados da conta! Este serviço é partilhado por todos os alunos.

### Exercício 8.16

Envie a página que acabou de criar para o servidor **xcoa.av.it.pt**, no diretório **~/public\_html**. Pode utilizar o comando **scp** ou o repositório **GIT** onde tem esta página.

Pode aceder à página através do endereço **http://xcoa.av.it.pt/~informatica-tXaY**, onde X é o número de turma e Y o número de aluno que lhe foi atribuído. Verifique que é possível adicionar comentários.

### Exercício 8.17

Aceda novamente ao endereço **http://disqus.com** e verifique as funcionalidades que lhe são apresentadas. Verifique que, por exemplo, ao adicionar comentários também pode ter informação sobre os seus visitantes e moderar os comentários.

## 8.6 Para aprofundar

### Exercício 8.18

Aceda à página do *Twitter Bootstrap* e implemente uma página local com todos os componentes extra que são apresentados.

### Exercício 8.19

Aceda à página do *Fuel UX* (<http://exacttarget.github.io/fuelux>) e implemente outros componentes que permitem enriquecer páginas Web.

## Glossário

<b>CSS</b>	Cascading Style Sheets
<b>HTML</b>	HyperText Markup Language
<b>JS</b>	JavaScript

## Referências

- [1] Google, *Google Mail*, <http://www.gmail.com>, [Online; acessado em 24 de janeiro de 2022], 2013.
- [2] ———, *Google Documents*, <http://doc.google.com>, [Online; acessado em 24 de janeiro de 2022], 2013.
- [3] M. Corporation, *Office - Office.com*, <http://office.microsoft.com>, [Online; acessado em 24 de janeiro de 2022], 2013.
- [4] W3C. (2001). «Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification», URL: <http://www.w3.org/TR/2011/REC-CSS2-20110607/>.
- [5] ECMA International, *Standard ECMA-262 – ECMAScript Language Specification*, Padrão, dez. de 1999. URL: <http://www.ecma-international.org/publications/standards/Ecma-262.htm>.
- [6] W3C. (1999). «HTML 4.01 Specification», URL: <http://www.w3.org/TR/1999/REC-html401-19991224/>.
- [7] H. JS, *Hicharts Demos*, <http://www.highcharts.com/demo/>, [Online; acessado em 24 de janeiro de 2022], 2013.
- [8] Google, *Google Maps*, <http://maps.google.com>, [Online; acessado em 24 de janeiro de 2022], 2013.
- [9] OpenStreetMap, *OpenStreetMap*, <http://www.openstreetmap.com>, [Online; acessado em 24 de janeiro de 2022], 2013.
- [10] V. Agafonkin, *LeafLet - a JavaScript library for mobile-friendly maps*, <http://leafletjs.com>, [Online; acessado em 24 de janeiro de 2022], 2013.
- [11] D. Inc., *Disqus - The Web's Community of Communities*, <http://www.disqus.com>, [Online; acessado em 24 de janeiro de 2022], 2013.